Weighted Transmedia Relevance Feedback for Image Retrieval and Auto-annotation

Thomas Mensink, Jakob Verbeek, Gabriela Csurka

▶ **To cite this version:**

Thomas Mensink, Jakob Verbeek, Gabriela Csurka. Weighted Transmedia Relevance Feedback for Image Retrieval and Auto-annotation. [Technical Report] RT-0415, INRIA. 2011. hal-00645608

HAL Id: hal-00645608
https://inria.hal.science/hal-00645608

Submitted on 28 Nov 2011

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Weighted Transmedia Relevance Feedback for Image Retrieval and Auto-annotation

Thomas Mensink, Jakob Verbeek, Gabriela Csurka

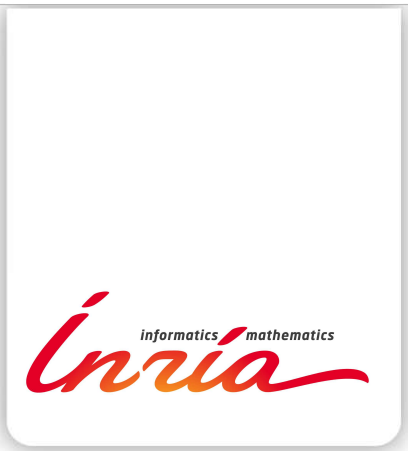
TECHNICAL REPORT

N° 0415

December 2011

Project-Teams LEAR - INRIA and TVPA - XRCE

ISSN 0249-0803 ISRN INRIA/RT--0415--FR+ENG



Weighted Transmedia Relevance Feedback for Image Retrieval and Auto-annotation

Thomas Mensink*†, Jakob Verbeek*, Gabriela Csurka†

Project-Teams LEAR - INRIA and TVPA - XRCE

Technical Report n° 0415 — December 2011 — 28 pages

Abstract: Currently large scale multimodal image databases have become widely available, for example via photo sharing sites where images come along with textual descriptions and keyword annotations. Most existing work on image retrieval and image auto-annotation has considered uni-modal techniques, either focusing on query-by-example systems or query-by-text systems for image retrieval, and mono modal classification for image auto-annotation. However recent state-of-the-art multimodal image retrieval and image auto-annotation systems combine different uni-modal models using late-fusion techniques. In addition, significant advances have been made by using pseudo-relevance feedback techniques, as well as using transmedia relevance models that swap modalities in the query expansion step of pseudo-relevance methods. While these techniques are promising it is not trivial to set the parameters that control the late fusion and pseudo/cross relevance models. In this paper, we therefore propose approaches to learn these parameters from a labeled training set: queries with relevant and non-relevant documents, or images with relevant and non-relevant keywords. Three additional contributions are the introduction of (i) two new parameterizations of transmedia and pseudo-relevance models, (ii) correction parameters for inter-query variations in the distribution of retrieval scores for both relevant and non-relevant documents, and (iii) the extension of TagProp, a nearest neighbor based image annotation method to exploit transmedia relevance feedback. We evaluate our models using public benchmark data sets for image retrieval and annotation. Using the data set of the ImageClef 2008 Photo Retrieval task, our retrieval experiments show that our learned models lead to significant improvements of retrieval performance over the current state-of-the-art. In our experiments on image annotation we use the COREL and IAPR data sets, and also here we observe annotation accuracies that improve over the current state-of-the-art results on these data sets.

Key-words: image auto annotation, multi-modal image retrieval, pseudo relevance feedback

* T. Mensink and J. Verbeek are in the LEAR group of INRIA Grenoble.
† T. Mensink and G. Csurka are in the TVPA group of Xerox Research Centre Europe.

RESEARCH CENTRE GRENOBLE – RHÔNE-ALPES

Inovallée
655 avenue de l'Europe Montbonnot
38334 Saint Ismier Cedex

Méthode de pertinence cross-modale pondérée pour le recherche et l'annotation d'images

Résumé : De nos jours, de plus en plus de larges bases d'images avec des textes et métadonnées associées sont disponibles sur la Toile. On peut mentionner par exemple, des sites de partage de photos où les images viennent avec des descriptions textuelles, des commentaires et des annotations avec des mots de clés. Malgré cela, les plupart des systèmes de recherche d'images existantes sont des techniques qui considèrent en générale une des modalités, soit visuelle, en se concentrant sur recherche par exemple, soit textuelle, en utilisant seulement le nom de l'image et les textes associés pour retrouver les images pertinentes. De même, les systèmes d'annotation automatique d'image sont en générale monomodales.

Cependant, de travaux récents sur la recherche d'images multimodales et annotations d'image ont montrés que la combinaison des différents systèmes uni-modaux, même par des techniques simples comme la fusion tardives des résultats, permet d'obtenir des résultats supérieurs à ceux obtenus avec des systèmes monomodaux.

En outre, des progrès significatifs ont été réalisés grâce à des techniques de type « pseudo-relevance feedback», notamment utilisant des modèles de pertinence. Ces modèles utilise d'abord une des modalités pour retrouver des éléments pertinents et pour enrichir la requête, puis une nouvelle recherche est effectué avec la requête enrichi, mais en utilisant l'autre modalité.

Bien que ces techniques sont prometteuses, ce n'est pas toujours triviale de définir les paramètres qui contrôlent les modèles de pertinence (pseudo et/ou cross) et leurs fusion. Dans ce papier, nous proposons donc des approches qui permettent d'apprendre ces paramètres à partir d'un ensemble d'apprentissage étiquetés, c'est-à-dire, des requêtes avec des documents pertinents et non pertinents ou des images avec des mots clés pertinents et non-pertinentes. Cette contribution du papier est complétée par : (i) l'introduction de deux nouvelles paramétrisations des modèles de pertinence pseudo et cross-modale ; (ii) la proposition des paramètres de correction des variations de la distribution des scores de pertinence d'une requête à une autre ; (iii) et l'extension de TagProp – une méthode d'annotation d'image basées sur la recherche des plus proches voisins – avec l'intégration des modèles de pertinence cross-modales.

Nos modèles sont évalués sur des données d'images de références publiques et souvent utilisées pour comparer les méthodes de recherche et d'annotation d'images. En utilisant l'ensemble des données de la tâche de recherche de photos pertinentes de l'ImageCLEF 2008, les résultats de nos expériences montrent que les modèles avec les paramètres apprises conduisent à des améliorations significatives des performances comparées à des résultats dans l'état de l'art actuel. De même, dans nos expériences concernant l'annotation d'image, utilisant les bases d'images COREL et IAPR, nous observons une amélioration des précisions d'annotation par rapport à la version de base de TagProp, qui est à ce jours parmi les méthodes les plus performantes sur ces ensembles de données.

Mots-clés : annotation automatique d'images, indexation et recherche multimodale d'images, modèle de pertinence pseudo et cross-modale

1 Introduction

The nature of today's multimodal databases indicates the need for techniques that effectively combine the different modalities present in documents. The great potential of exploiting the relation between the different modalities can be understood by viewing the different modalities in a document as forms of mutual, albeit noisy, supervision. Recently, this idea has been leveraged to learn recognition models from little or no manual supervision, for example for face recognition based on image captions [7, 23], accurate object recognition models learned from noisy search engine results [51, 29], and action recognition in video using script alignment in combination with text classification to obtain noisy annotation of video clips [30].

In this paper we are interested in document retrieval systems and document annotation systems for multimodal image databases, where different types of meta-data are associated with each image, *e.g.* a caption, a set of keywords, or location information. Examples of such databases include: image sharing websites showing comments, location and keywords for images (*e.g.* `www.flickr.com`, or `picasaweb.google.com`), news websites publishing articles with photographs (*e.g.* `news.bbc.co.uk`, or `news.yahoo.com`), and corporate collections of publications and presentations with pictures, graphs, and text.

Most of the current retrieval methods, however, follows one of two paradigms: query-by-text, or query-by-example. Internet image search engines typically use query-by-text, the user provides a textual description of the target images, and the text in metadata and around images on websites are used to rank the images. In query-by-example, the user provides an example image and visual similarity is used to rank the database images according to their relevance, recently this has been scaled to large databases [25]. Image annotation can also be seen as a query-by-example retrieval system, where the query image is labeled with the tags of its visual nearest neighbors [37, 22, 39].

1.1 Multimodal document retrieval

Multimodal document retrieval differs from text retrieval in the sense that both the query and the documents can contain multiple modalities, *e.g.* a text with an image. It has been observed that combining text-based and image-based retrieval methods can significantly improve retrieval performance on multimodal databases [41].

Different modalities are typically combined using late or early fusion. Late fusion models use a separate retrieval model for each modality and use a weighed sum of these scores to produce a final score, where the weights could be fixed or learned. These models are attractive because of their simplicity, however they do not exploit the correlations between different modalities, and cannot be used to match queries and a documents in different modalities. Early fusion models in contrast, attempt to exploit the correlations between the different modalities at feature level. The main challenge is to find a joint representation which allows for variations in the modalities due to *e.g.* the level of semantic meaning (words vs. low level image features), their difference in dimensionality, and to handle the absence of a modality. Comparisons between early and late fusion methods lead to different conclusions, for image and video classification tasks the best strategy seems to depend on the specific tasks and data set [28, 52].

Intermediate level fusion methods, also known as cross-media or transmedia models, generalize the pseudo-relevance feedback principle, where the used modality is swapped at the query expansion step [10, 36, 13]. For example, for a visual input

query, visual similarities are used to select the set of most similar documents to this query. In the query expansion step the textual modality of these documents are used to rank all documents in the database. They have shown excellent performance in multimodal retrieval tasks, *e.g*. ImageClef Retrieval evaluations [41, 4, 2]. As they exploit the correlation between the visual and textual modality of a document, the provided cross-modal retrieval system is able to handle naturally documents or queries with only a single modality, and therefore it is also suitable for tasks such as image auto-annotation [39].

1.2 Image annotation

The goal of auto-annotation is to predict relevant keywords from a finite vocabulary for a given image. A popular approach to auto-annotation is to learn keyword-specific binary image classifiers for all terms in the vocabulary. While this yields state of the art performance in many classification tasks [17], this might be costly in case of large and dynamic image sets, and few of the actual systems scale well to large amount of classes.

An alternative solution is tag propagation: to annotate an image, similar images are gathered from the training data set and the annotations are deduced from analysing the annotations of the retrieved images. One can either directly deduce the most relevant concepts/keywords to tag the test image [34, 55] or learn a discriminative model in neighborhoods of test images [59]. Recently, these nearest neighbor type methods have shown excellent performance for auto-annotation [37, 22, 38]. In this paper we extend TagProp [22], a nearest neighbor model that predicts tags by taking a weighted combination of the tag absence/presence among the neighbors. In contrast to [22], where the neighbors are found purely on visual information, we also use the available textual information around the database images by integrating the transmedia pseudo-relevance feedback [36, 13] approach.

1.3 Goal and outline of the paper

In this paper we study approaches of learning transmedia relevance methods for multimodal document retrieval and image auto-annotation. Transmedia relevance models require a number of parameters to be set: the number of documents used in the query extension step, and the late fusion weights of a combination of retrieval models. We frame the transmedia relevance models in a probabilistic framework, which allows us to learn the parameters automatically from training data, allowing us to combine larger numbers of retrieval scores. Our experiments on mutli-modal document retrieval (using the ImageClef 2008 Photo Retrieval task) and image annotation (using the Corel 5k and IAPR TC12 data sets), show that our models outperform current state-of-the-art results.

The contributions we present in this paper are:

1. We explore two parameterisations of transmedia relevance and pseudo-relevance feedback models for image retrieval and image annotation.

2. We introduce a method to correct for inter-query variations in the distribution of retrieval scores for the learning of retrieval functions.

3. We compare two models to learn the parameters of retrieval functions from training queries with documents labeled by relevance.

4. We extend TagProp to use our transmedia relevance models for image auto-annotation.

This paper extends our earlier work [39] on learned transmedia relevance models for image annotation, to multimodal document retrieval (Section 4). Furthermore, we present a more extensive review of related work, and additional experimental results.

In Section 2 we provide some further background on multimodal document retrieval and image auto-annotation methods which are closest to and inspired our methods. In Section 3 we recall the principles of pseudo-relevance feedback and transmedia relevance models, and propose different parametrizations for them. In Section 4 we present our approaches to learn these parameters for the retrieval models, and introduce the query correction terms. In Section 5 we describe the extension of TagProp with the transmedia relevance models for image auto-annotation. In Section 6 we present experimental results both for retrieval and image auto-annotation, and we conclude our paper in Section 7.

2 Related work

Currently there are large collections of documents available that contain multiple modalities, such as image sharing websites providing tags, location and comments on an images, news websites providing illustrated news articles, and more generally any webpage containing an image with associated text. The multimodal nature of these databases suggests that the methods to access these data, *e.g*. using clustering, classification, retrieval and visualization tools, should exploit the multimodality. An advantages of using multiple modalities is that it can be seen as a form of weak supervision, which is cheap to obtain [6, 23]. While it is usually straightforward to obtain multimodal data, it is often expensive to create a large manually annotated multi media databases. In order to exploit multimodal data sets, the main challenge is to model the often noisy relation between the modalities.

2.1 Multimodal document retrieval

Retrieval systems for multimodal documents use still mostly only one of the available modalities, either the text or the image. However, methods that combine different modalities can improve retrieval performance on multimodal databases [41, 4, 2]. These methods can be roughly divided in ones that use late fusion or early fusion models.

Late fusion models are attractive due to their simplicity. For each modality a separate retrieval function is obtained, the final score for a document is the weighted sum of these scores. The combination can be done by a simple averaging of ranking scores from models learned on the different media, or by learning an additional fusion model (*e.g*. a linear combination, or more complex functions) that depends on the uni-modal scores. Despite their simplicity late fusion models have the disadvantage they cannot exploit the correlations between the different modalities, since each modality is treated independently. Furthermore, they can only handle query and document pairs which have the same modality, *i.e*. they can not assign a relevance score for a visual document given a textual query.

Early fusion models attempt to exploit the correlations between the different modalities by finding a joint representation, an example is topic models that have been used

for image annotation [5]. The joint representation should allow for the heterogeneity of the different modalities, due to variations in their level of semantic meaning (words vs. low level image features), and due to different dimensionalities. Different authors have found mixed results when comparing early and late fusion methods for image and video classification tasks [52, 28]; the best strategy seems to vary across the different classification tasks. Similarly, for multi-modal document retrieval late fusion is the most used approach, but early fusion techniques are (sometimes) found to be better depending on the data and the queries [15].

Recently, several authors have proposed methods for multimodal retrieval that may be understood as performing intermediate fusion, which are known as cross-media or transmedia relevance models [10, 36, 2]. These models are based on pseudo-relevance feedback, but swap modalities at the query expansion step. Pseudo-relevance feedback models were originally developed in the context of text retrieval [50], but have also been successfully used in image retrieval systems [12]. The main idea is to query the database with an extended query consisting of (1) the initial query, which contains usually only of a small number of words, and (2) text taken from the most relevant documents to the initial query. This can improve retrieval performance since the new query is likely to contain words related to the original query terms, and therefore a more robust matching is obtained.

In transmedia relevance models the similarity functions used in the two retrieval steps are based on different modalities. For example, we start with a query image and select the k most similar images from the database based on visual similarity. Then, the text associated with these k images is used to re-rank the documents according to their textual similarity to them. These models have shown significant improvement on retrieval performance in multimodal databases [2].

2.2 Image annotation

Image annotation is a well studied topic in computer vision research. Due to the vast amount of literature, we discuss only the most relevant image annotation and keyword based retrieval models for our work.

Topic based models such as latent Dirichlet allocation, probabilistic latent semantic analysis, or hierarchical Dirichlet processes, have been explored for image annotation by several authors [5, 40, 57]. They model the annotated images as samples from a mixture of topics, where each topic is a distribution (most often Gaussian) over image features and annotation words (generally multinomial). The mixing weights vary per image and have to be inferred using sampling or variational EM procedures. Methods inspired by machine translation [16], where visual features are translated into the annotation vocabulary, can also be seen as topic models, where one topic is used per visual descriptor type. Although conceptually attractive, their expressive power is limited by the number of topics.

A second family of methods uses mixture models to define a joint distribution over image features and annotation tags [26, 31, 18, 9]. As each training image is used as a mixture component, these models can be seen as non-parametric density estimators over the co-occurrence of images and annotations. To annotate a new image, these models compute the conditional probability over tags given the visual features by normalizing the joint likelihood. As above, generally Gaussian mixtures are used to model visual features, while the distributions over annotations are multinomials or separate Bernoullis for each word.

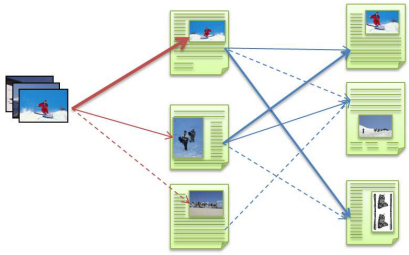


Figure 1: Schematic illustration of ranking documents using transmedia relevance models. The set of documents most similar to the query is obtained using their visual similarity. Then the textual parts of these documents are used to re-rank all documents in the database using textual similarity.

Both families of generative models are critisized because maximizing the generative data likelihood might not be necessarily optimal for predictive performance. Therefore, alternatively, discriminative models for tag prediction were proposed in [24, 14, 20] that learn a separate classifier for each potential tag. This is equivalent to multi-class multi-label image classification problem, and hence different learning methods can been used to train the classifiers, including support vector machines, Bayes point machines, *etc*.

Given the increasing amount of training data that is currently available, local learning techniques are becoming more attractive as a simple yet powerful alternative to parametric models. Examples of such techniques include methods based on label diffusion over a similarity graph of labeled and unlabeled images [35, 45], or learning discriminative models in neighborhoods of test images [59]. A simpler ad-hoc nearest neighbor tag transfer mechanism was introduced in [37], showing state-of-the-art performance by combining distances based on a set of diverse image features into a powerful combined distance. As a generalization of this method, Guillaumin *et al*. in [22] proposed TagProp that learns the weights for each neighbor by maximizing the likelihood of annotations in a set of training images. In this paper, we propose to integrate transmedia pseudo-relevance models within TagProp to improve image auto-annotation performance.

Image annotation and classification has also been studied in a multimodal setting. For example in [38, 54], SVM classifiers and TagProp are used in a setting where besides the image also user provided tags are available during training and testing. The use of the visual and textual modality significantly improves the results over using the visual or textual modality alone. The setting in our paper is different, since we assume to have a multimodal training database, but during test time we use only the image as input.

3 Pseudo-relevance feedback

Relevance feedback models were originally developed in the context of text retrieval [50]; it is a query expansion method where a user selects relevant documents from an initial ranking to extend the query. The extended query is used to improve retrieval performance.

Pseudo-relevance feedback models automatise relevance feedback; instead of relying on user feedback, the top k retrieved documents are assumed to be relevant and used to enrich the original query. For example in text retrieval, the most frequent words of the top k documents are added to the original query and used to obtain the final document ranking. This often improves retrieval performance since user queries tend to be

Table 1: Overview of the different direct, pseudo-relevance, and transmedia similarities using visual and textual modalities.

query	text	image
direct	$s_t(q, d)$	$s_v(q, d)$
pseudo-relevance	$s_{tt}(q, d)$	$s_{vv}(q, d)$
transmedia	$s_{tv}(q, d)$	$s_{vt}(q, d)$

too short to precisely capture the user intention, while the extended query is likely to contain words related to the original query terms, and therefore a more robust matching is obtained.

3.1 Transmedia relevance models

Transmedia relevance models are the multimodal generalisation of pseudo-relevance feedback models, where the modality is swapped in the query extension step [10, 36, 2]. For example, in the first retrieval step a visual similarity is used, while for the second step a textual similarity is used, see Fig. 1 for a schematic illustration. Such models are useful for databases where at least some of the documents contain both a visual and textual data.

One of the advantages of such models over early fusion approaches is that both query expansion and document ranking are based on single modalities, although in combination multiple modalities are exploited. This allows the use of well engineered mono-modal retrieval systems, *e.g.* specifically developed for text or image retrieval. These approaches go beyond simple late fusion of mono-modal retrieval functions, exploiting the link between the visual and textual content as encoded by the co-occurrences of these in the multimodal documents in the database.

Another advantage of these models over both late and early fusion models is that multimodal, uni-modal, and cross-modal queries are all handled naturally. The system can exploit the multimodal documents in the data to return pure-text documents that are relevant for pure-image queries. For multimodal queries the different modalities bring complementary information.

In this paper we focus on documents and queries containing visual and textual modalities, however the presented methods also hold for other (combinations of) modalities. We assume that for each modality a similarity measure is available, denoted by s_v and s_t for the visual and textual modality respectively, in Section 6 we describe the visual and textual similarities we have used. As a baseline we will use the model of [1], which defines the transmedia similarity between a query q and a document d as

$$s_{ab}(q, d) = \sum_{i=1}^{k} s_a(q, d_i)\ s_b(d_i, d), \tag{1}$$

where a and b denote the used modalities, and d_i denotes the i-th most similar document to q according to s_a. While instead of Eq. 1, we could use only the aggregation $\sum_{i=1}^{k} s_b(d_i, d)$ of *e.g.* textual scores, the weighting with the score $s_a(q, d_i)$ allows to give higher importance to documents that are both both visually and textually similar.

An overview of direct, pseudo-relevance, and transmedia similarities are shown in Table 1. Below, we define retrieval functions to combine these different models for

image retrieval Section 4, and in Section 5 we discuss how to extend TagProp with these models for image annotation.

3.2 Parametrized relevance feedback models

In this section, we present two parametrized alternatives for Eq. 1. In the first parametrization, we associate a fixed, rank-based weight with each of the k most similar documents. Using γ_i as the weight for the i-th neighbor, we define

$$s_{ab}(q,d) = \sum_{i=1}^{k} \gamma_i \; s_a(q,d_i) \; s_b(d_i,d). \tag{2}$$

Clearly, Eq. 1 is a special case of this model, where $\gamma_i = 1/k$ for all k neighbors. Since we expect a positive relation between the neighbor similarities and the final similarity, we can impose a non-negativity constraint on the coefficients γ_i of the linear combination. Further, neighbors are ordered on their distance and so we expect that the weight of neighbor $i+1$ should not exceed that of neighbor i, *i.e.* $\gamma_i \geq \gamma_{i+1}$. Both constraints allow for better generalization, and they define a convex set of feasible γ_i values.

The second model we use satisfies the non-negativity and ordering constraints by construction. We use the softmax function on $s_a(q,d')$ to define the following weighting over the neighboring documents:

$$s_{ab}(q,d) = \sum_{i=1}^{k} \tilde{s}_a(q,d_i) \; s_b(d_i,d), \tag{3}$$

$$\tilde{s}_a(q,d_i) = \frac{\exp(\gamma \; s_a(q,d_i))}{\sum_{j=1}^{k} \exp(\gamma \; s_a(q,d_j))}. \tag{4}$$

This model has the advantage that it only has a single parameter γ as opposed to the k parameters $\{\gamma_i\}_{i=1}^{k}$ in Eq. 2. Furthermore, the weights assigned to neighbors vary smoothly with the similarity, which might be beneficial. Consider for example two documents d_i and d_j that almost have the same similarity to a query, *e.g.* $s_a(d_i,q) = s_a(d_j,q)+\epsilon$ for a small ϵ. In this example the rank-based weights of the two documents can change drastically depending on γ_i, γ_j, and the sign of ϵ. On the other hand, the weights of the two documents will remain close according to Eq. 4, as desired.

In the following sections we will use these transmedia relevance formulations in retrieval and annotation models.

4 Learning score functions for multimodal retrieval

Learning to rank is an active area of research, which is partly due to its relevance for tuning web search engines such as Google, Yahoo and Bing, and the availability of large data sets, *e.g.* the LETOR dataset [49]. Most work focuses on either text retrieval or image retrieval, however most of the proposed approaches are independent of which modality is used, *i.e.* they can be used both for text or image retrieval. Different approaches that have been proposed include, using a probabilistic cost function [8], SVM-based methods [27, 19], online Passive-Aggressive discriminative learning [20], and recently a focus on large-scale (image) datasets emerges [11, 56]. For an extensive overview see [33].

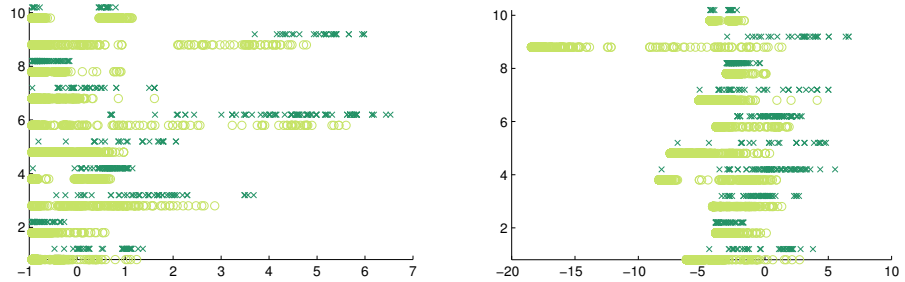


Figure 2: Effect of the query dependent correction terms α_q and β_q when learning the relevant classification (RC) model. For ten queries (organized vertically) we show the distribution of $f(q, d)$ for relevant documents ($\times$) and non-relevant ones ($\circ$). In the left panel we do not use the correction terms, yielding $\mathbf{w} = [3.1, 2.7]^\top$, while in the right panel we include them and obtain $\mathbf{w} = [5.1, 9.1]^\top$.

We define the retrieval function f as a linear function of the various mono-modal and multimodal similarities presented in the previous section and summarized in Table 1. Let $\mathbf{x}_{qd}$ denote the vector that collects these similarities for a query q and a document d. Then

$$f(q, d) = \mathbf{w}^\top \mathbf{x}_{qd} + w_0. \tag{5}$$

where $\mathbf{w}$ and w_0 are parameters that control the late fusion of these similarities.

We learn the parameters of this function using training data which consists of queries q with corresponding lists of relevant and non-relevant documents, denoted R_q and N_q respectively for a query q. In practice, they might be automatically obtained from *e.g.* click-through logs, although the labeling would be noisy in that case.

We consider two classification models to learn the parameters $\mathbf{w}$ of (5), and the internal parameters to define the similarities $\mathbf{x}_{qd}$. The first, is a document classifier which directly tries to label the document as relevant or non-relevant for a given query. The second, is a comparative classifier which predicts the most relevant document from a pair of a relevant and a non-relevant document. Both models have been explored in the context of text retrieval [27, 32], and it has been shown that these models could be complementary [60]. However, learning the trade off between the two losses is non-trivial, and requires *e.g.* cross-validation on the used data sets, therefore we do not consider this combination. To the best of our knowledge the two models were not compared before to learn transmedia relevance feedback models.

Next, in Section 4.1, we present the two objective functions to learn the retrieval models, and in Section 4.2 we introduce correction terms that can account for inter-query variations in the retrieval scores. Finally, in Section 4.3 we discuss how we implement the learning algorithms in practice.

4.1 Objective functions to learn retrieval models

In this section we formalize the two learning objectives we use to learn the parameters of the late fusion function (5) and the internal parameters of the similarities, *i.e.* the parameters controlling the pseudo/transmedia relevance feedback models.

4.1.1 Relevance Classification (RC)

This model defines a binary classification problem where query-document pairs have to be classified as relevant or non-relevant [32]. We use $y_{qd} \in \{-1, +1\}$ to denote the class label (non-relevant or relevant) of a query-document pair, and define the class probabilities using the logistic discriminant model as:

$$p(y_{qd} = +1) = \sigma(f(q, d)) = \sigma(\mathbf{w}^\top \mathbf{x}_{qd} + w_0), \tag{6}$$

where σ is the sigmoid function, $\sigma(z) = (1 + \exp(-z))^{-1}$. The objective is to maximize the log-likelihood of correct classification of all query-document pairs:

$$\mathcal{L}_{RC} = \sum_q \sum_{d \in R_q} \ln p(y_{qd}{=}1) + \sum_{d \in N_q} \ln p(y_{qd}{=}\text{-}1) \quad = \sum_q \sum_{d \in D} \ln \sigma(y_{qd}\, f_{qd}). \tag{7}$$

The objective function is concave in $\mathbf{w}$ and w_0, and can be optimized for example using gradient-based methods. Let θ denote the set of parameters we optimize for, then the derivative of $\mathcal{L}_{RC}$ *w.r.t.* θ is given by

$$\frac{\partial \mathcal{L}_{RC}}{\partial \theta} = -\sum_q \sum_{d \in D} y_{qd}\, \sigma(-y_{qd}\, f_{qd})\, \frac{\partial f_{qd}}{\partial \theta}. \tag{8}$$

4.1.2 Comparative Classification (CC)

Learning a classifier to predict document relevance might not be optimal if the goal is to perform ranking on the score function. Instead, we can learn a score function based on pair-wise comparisons, that tries to ensure that each relevant document has a larger score than each non-relevant document. To this end, we define a classification problem over pairs consisting of a relevant document $d \in R_q$ and an non-relevant one $d' \in N_q$. Using the relevance labels y_{qd} and $y_{qd'}$ as before, the goal is to predict which of the two documents is the relevant one and which is the non-relevant one:

$$p(y_{qd} > y_{qd'}) = \sigma(f(d, q) - f(d', q)). \tag{9}$$

The objective in this case is to maximize the log-probability of correct classification of all pairs for each query:

$$\mathcal{L}_{CC} = \sum_q \sum_{d \in R_q} \sum_{d \in N_q} \ln p(y_{qd} > y_{qd'}). \tag{10}$$

As before, the model is concave in $\mathbf{w}$ and w_0, and can be optimized using gradient-based methods. The derivative of $\mathcal{L}_{CC}$ *w.r.t.* the parameters θ is given by

$$\frac{\partial \mathcal{L}_{CC}}{\partial \theta} = -\sum_q \sum_{d \in R_q} \sum_{d' \in N_q} \sigma(f_{qd'} - f_{qd}) \left(\frac{\partial f_{qd}}{\partial \theta} - \frac{\partial f_{qd}}{\partial \theta} \right), \tag{11}$$

where $f_{qd} = f(q, d)$. A similar model was used in [27], albeit using a hinge loss instead of a logistic loss.

4.2 Correcting for inter-query variations

The above objective functions to learn the parameters are defined by summing log-likelihood functions for different queries. In practice we encounter large differences in the distribution of similarities for relevant and non-relevant documents across different queries. Not only the mean similarity to the query from the relevant documents changes significantly, but also the variance of these similarities varies significantly across queries.

For the ranking performance this does not pose any problem, since ranking is invariant to additive and (positive) multiplicative variations of the scoring function. The objective functions defined above, however, are not invariant to these transformations. The problem is that the objective function optimizes macro-precision —a single cut-off point used for all queries— while we are interested in maximizing micro-precision where a different cut-off point is used per query [47].

To render the objective functions invariant to additive and multiplicative constants per query, we redefine the retrieval function as:

$$f(q,d) = \alpha_q \mathbf{w}^\top \mathbf{x}_{qd} + \beta_q, \tag{12}$$

where α_q, β_q are free parameters in the optimization. Using Eq. (12) the objective functions $\mathcal{L}_{RC}$ and $\mathcal{L}_{CC}$ become bi-concave, given $\mathbf{w}$ they are concave in $\boldsymbol{\alpha}$, and $\boldsymbol{\beta}$, and given $\boldsymbol{\alpha}$ they are concave in $\mathbf{w}$ and $\boldsymbol{\beta}$. It is important to note that we do not need α_t and β_t for a test query, since the ranking according to $f(q,d)$ equals the ranking according to $\mathbf{w}^\top \mathbf{x}_{qd}$.

The idea of using a correction bias (β_q) is independently from our work introduced in [3] for its use in text retrieval on the LETOR[49] dataset. Ailon calls it the '*intercept*' with the intuition to allow for different relevance criterion for different queries. In our experiments we observe that not only the bias is query dependent, but also the scaling of the similarity scores, which is corrected for by α_q. In Fig. 2 we illustrate the effect of learning the α_q and β_q parameters in practice, which shows that without these terms it is difficult to find a single cut-off on the score to discriminate relevant and non-relevant documents. When the correction terms are included the retrieval scores $f(q,d)$ for relevant and non-relevant documents across different queries become more comparable. Importantly, note that the learned values for late fusion are qualitatively different: without the correction terms more weight is given to the first similarity measure, while the situation is reversed when correction terms are used.

4.3 Implementation of the learning algorithms

The input of the learning algorithm is a set of training queries, for each of which a set of relevant and non-relevant documents is provided. The learning algorithm then maximizes the objective function ($\mathcal{L}_{RC}$ or $\mathcal{L}_{CC}$), using gradient ascent. See Algorithm 1 for an overview of the learning procedure.

When using the rank-based formulation of the transmedia feedback models from Eq. 2, the transmedia similarity $s_{ab}(q,d)$ is a linear function of γ_i. Therefore we can absorb γ_i into $\mathbf{w}$ and $s_a(q,d_i)s_b(d_i,d)$ into the vector $\mathbf{x}_{qd}$. For the transmedia similarity ab this means that w_{ab} and x_{ab} are redefined to:

$$\mathbf{w}_{ab} = [\gamma_1, \ldots, \gamma_k], \tag{13}$$

$$\mathbf{x}_{ab} = [s_a(q,d_1)s_b(d_1,d), \ldots, s_a(q,d_k)s_b(d_k,d)]. \tag{14}$$

while *not converged* **do**
  maximise $\mathcal{L}$ *w.r.t.* γ (when using softmax Eq. 4);
  maximise $\mathcal{L}$ *w.r.t.* $\mathbf{w}$ (and $\{\beta_q\}$ for $\mathcal{L}_{RC}$);
  maximise $\mathcal{L}$ *w.r.t.* $\{\alpha_q\}$ (and $\{\beta_q\}$ for $\mathcal{L}_{RC}$);
  calculated log-likelihood with current parameters;
  check for convergence;
end

Algorithm 1: Iterative learning of ranking functions for $\theta = \{\mathbf{w}, \gamma, \{\alpha_q\}, \{\beta_q\}\}$.

Thus, we directly learn a linear weighting of neighbors for transmedia feedback, while also combining several mono-modal or multimodal similarity measures.

When using the softmax weighting for transmedia feedback as in Eq. 4, we iteratively optimize over γ for fixed $\mathbf{w}$, and w_0, and then over $\mathbf{w}$ and w_0 for fixed γ. The optimization over γ is not convex, and we use an approximate second order gradient ascent method.

If we include the correction terms α_q, β_q, the learning objective functions remain as before, except that we will now maximize them over the generic linear combination $\mathbf{w}$, but also over the query specific auxiliary variables $\{\alpha_q, \beta_q\}$. Since the score function is now bi-linear in the parameters, we optimize it in alternation over $\mathbf{w}$ and the α_q, which is a convex problem in both cases. The β_q parameters are optimized jointly with both $\mathbf{w}$ and the α_q. Note that the bias terms w_0 and β_q, are only useful in the Relevance Classification model, since they cancel out in the Comparative Classification model.

For the optimization of $\mathcal{L}_{CC}$ and $\mathcal{L}_{RC}$ we still need the derivatives of f_{qd} *w.r.t.* the parameters θ. The derivatives for $\mathbf{w}, \alpha_q$, and β_q are trivial. The derivative for γ_{ab} that controls the softmax that combines similarities measures s_a and s_b, is given by

$$\frac{\partial f_{qd}}{\partial \gamma_{ab}} = w_{ab} \sum_q \alpha_q \sum_{i=1}^{k} \tilde{s}_a(q, d_i) s_b(d_i, d) \left(s_a(q, d_i) - \sum_{j=1}^{k} \tilde{s}_a(q, d_j) s_a(q, d_j) \right), \tag{15}$$

where w_{ab} denotes the corresponding entry in $\mathbf{w}$.

The number of parameters we have to learn scales linearly in the number of queries (for α_q, β_q), plus a fixed number for the late-fusion weights and the pseudo-relevance components. In our best performing model (see Section 6.1) we use the softmax-weighting, with 2 direct models and 4 relevance feedback models, for this model we have to learn 52 parameters: (a) a single $\mathbf{w}$ vector for late fusion of the 6 components (6 parameters); (b) a γ scaler for each of the 4 softmax relevance feedback model (4 parameters); and (c) a α_q and β_q per train query - 21 in our experiments, (42 parameters).

5 Image auto-annotation

In this section we will apply our parametrized transmedia relevance models for image annotation. To do so we embed them in TagProp, a recent state-of-the-art image annotation method based on nearest-neighbor prediction [22]. The idea is to use the transmedia relevance model to define nearest neighbors, using visual similarity to the image to be annotated and textual similarity among the training images.

In the next subsection we briefly describe TagProp, and in Section 5.2 we present how we include the transmedia relevance feedback model.

5.1 Image annotation with TagProp

To model image annotations, TagProp uses a Bernoulli distribution to model the absence/presence of each keyword for an image. Let $y_{it} \in \{-1, +1\}$ denote the absence/presence of keyword t for image i, hence encoding the image annotations. The presence prediction $p(y_{it} = +1)$ for keyword t from image i is defined as a weighted sum over the training images, indexed by j:

$$p(y_{it} = +1) = \sum_j p(y_{it} = +1|j)\, p(j|i), \tag{16}$$

$$p(y_{it} = +1|j) = \begin{cases} 1 - \epsilon & \text{if } y_{jt} = +1, \\ \epsilon & \text{otherwise,} \end{cases} \tag{17}$$

the ϵ is a technicality to avoid zero prediction probabilities, in practice we set $\epsilon = 10^{-5}$.

The probability to use image j as a neighbor for image i, $p(j|i)$, can be defined based on the rank of image j in the list of nearest neighbors of image i, or directly based on the distance between image i and j. While the performance does not depend much on this choice [54], we prefer the distance based interpretation. This interpretation has the advantage that the weights depend smoothly on the distance, which allows for metric learning. In this case, using $\mathbf{d}_{ij}$ to denote a vector of various distances between image i and j, we define

$$p(j|i) = \frac{\exp(-\mathbf{w}^{\mathrm{T}}\mathbf{d}_{ij})}{\sum_{j' \in \mathbf{J_i}} \exp(-\mathbf{w}^{\mathrm{T}}\mathbf{d}_{ij'})}. \tag{18}$$

where the vector $\mathbf{w}$ takes a linear combination of the different distances, and $\mathbf{J_i}$ can be the complete training data set, or the subset of the J most similar images to i in which case images outside $\mathbf{J_i}$ are assigned zero weight.

To estimate the parameter vector $\mathbf{w}$, the log-likelihood of the predictions of training annotations is maximized. Taking care to set the weight of training images to themselves to zero, *i.e.* $p(i|i) = 0$, the objective is to maximize

$$\mathcal{L} = \sum_i \sum_t c_{it} \ln p(y_{it}), \tag{19}$$

where c_{it} is a weight that takes into account the imbalance between keyword presence and absence. If $y_{it} = +1$ we use $c_{it} = \frac{1}{n^+}$, where n^+ is the total number of keyword presences, and similarly we use $c_{it} = \frac{1}{n^-}$ for $y_{it} = -1$. This weighting is used since in practice there are many more keyword absences than presences, and absences are much noisier than presences because often images are annotated with only a subset of all possible relevant keywords.

In [22] an extended model is proposed which uses word-specific logistic discriminant models to boost recall performance. However, in general the performance on mean average precision (MAP) is for both models almost identical[54]. Therefore, in this paper we consider the former one as baseline and we compare our method to it, nevertheless our proposed extension can easily be integrated with the word-specific models.

while *not converged* **do**
  minimize log-likelihood $\mathcal{L}$ *w.r.t.* γ parameters
  compute d_{vt} given the γ parameters
  minimize $\mathcal{L}$ *w.r.t.* $\mathbf{w}$ using distances $\mathbf{d}_{ij}$
  check for convergence of the log-likelihood $\mathcal{L}$.
end

Algorithm 2: Optimizing TagProp using softmax transmedia feedback models.

5.2 Extension with transmedia relevance

We now show how we include the transmedia pseudo-relevance feedback model of Section 3 into TagProp. We have defined transmedia score between a query and a document using Eq. 2, and Eq. 4. In short they allow us to define a visual-to-textual dissimilarity as

$$d_{vt}(i,j) = \sum_k \hat{d}_v(i,k)\, d_t(k,j), \tag{20}$$

$$\hat{d}_v(i,k) = \begin{cases} \gamma_k\, d_v(i,k) & \text{linear model Eq. 2,} \\ \frac{\exp(-\gamma\ d_v(i,k))}{\sum_{k'} \exp(-\gamma\ d_v(i,k'))} & \text{softmax model Eq. 4.} \end{cases} \tag{21}$$

While these models are similar to Section 3, here a dissimilarity measure between two documents is defined, and not a similarity. The new transmedia distance of Eq. 21 can replace the distance vector $\mathbf{d}_{ij}$ in Eq. 18, or it can be added to the vector of distances with an additional weight.

5.3 Learning the parameters of the model

TagProp is optimized using a projected gradient algorithm to directly maximize the log-likelihood function. The gradient of Eq. 19 *w.r.t.* the general parameters θ equals

$$\frac{\partial \mathcal{L}}{\partial \theta} = \sum_{i,j} \left(C_i p(j|i) - \sum_t c_{it} p(j|y_{it}) \right) \frac{\partial \mathbf{w}^{\mathrm{T}} \mathbf{d}_{ij}}{\partial \theta}, \tag{22}$$

where $C_i = \sum_t c_{it}$. To reduce the computational cost, only the pairwise distances over a large set of J neighbors is used, the remaining ones are assumed to be zero.

In the case that linear transmedia models are used, each neighbor k from the first step is used as separate distance, and we absorb the γ_k parameters in the the vector $\mathbf{w}$. This allows the direct use of the original TagProp gradient Eq. 22, with $\frac{\partial \mathbf{w}^{\mathrm{T}} \mathbf{d}_{ij}}{\partial \theta} = \mathbf{d}_{ij}$, where $\mathbf{w}$ is the extended parameter vector.

When using the softmax transmedia models, we optimize for γ and $\mathbf{w}$ iteratively, as described in Algorithm 2. The derivative of Eq. 19 *w.r.t.* γ_d equals Eq. 22 using:

$$\frac{\partial \mathbf{w}^{\mathrm{T}} \mathbf{d}_{ij}}{\partial \gamma_d} = w_d \sum_k \tilde{d}_v(i,k) d_t(k,j) \left(d_v(i,k) - \sum_{k'} \tilde{d}_v(i,k') d_v(i,k') \right). \tag{23}$$

Here we described the models using a single transmedia component, extensions using multiple parameterized transmedia components are straightforward.

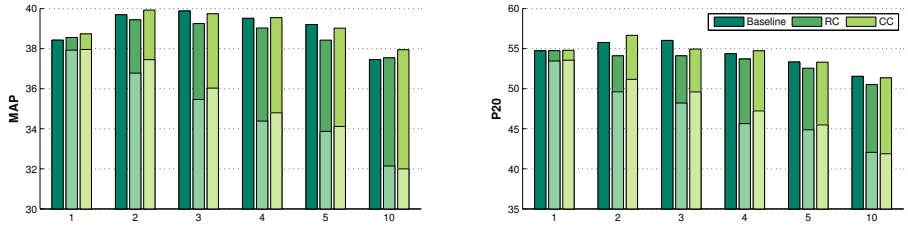


Figure 3: MAP (left) and P20 (right) of the baseline and our models, for several values of k. For the learned classification models, we show the results without (lighter bars) and with (darker bars) the query specific correction terms $\{\alpha_q, \beta_q\}$.

6 Experimental results

In this section we describe our experimental evaluation of the proposed transmedia models for retrieval (Section 6.1) and image annotation (Section 6.2).

6.1 Retrieval experiments

In this section we experimentally evaluate our models on the IAPR TC-12 data set [21]. This data set has been used in the ImageCLEF photo retrieval task of 2007 and 2008, where documents have to be ranked given a query consisting of a short caption and three images. The performance is measured by evaluating the system over different queries and calculating the mean average precision (MAP) and the mean precision at 20 (P20). From the 60 queries used in 2007, 39 were re-used in the challenge of 2008. We take those 39 queries as test set, and use the remaining 21 queries as train set. Some typical test queries are given in Fig. 4, where we also show the top 10 retrieved images using our method.

6.1.1 Image and text similarities

The image and text features we use for our experiments are similar to the features described in [1]. As image representation, we use the Fisher vectors [46], extracted separately from two channels, one for texture using local SIFT descriptors, and the other for color features. The resulting two vectors are concatenated to obtain a single image signature. The similarity between the signatures f_d and $f_{d'}$ of two images is defined as 2 minus their negative ℓ_1 distance:

$$s_v(d, d') = 2 - \sum_k |f_d^{(k)} - f_{d'}^{(k)}|. \tag{24}$$

For the textual representation we use a probabilistic language modeling approach to information retrieval on pre-processed texts [48]. The pre-processing includes tokenization, lemmatization, word de-compounding and a standard stop-word removal. The word counts associated with each document are obtained by adding counts from the title, the location, and the description fields of each document. As similarity measure between two documents we use the cross-entropy:

$$s_t(d, d') = \sum_w p(w|d) \log p(w|d'). \tag{25}$$

Table 2: In this table we show a comparision of the results for different definitions of the transmedia component s_{vt}, combined with s_t by late fusion, using the $\mathcal{L}_{CC}$ model. For each value of k the best results are highlighted bold.

	Eq. Weighed		**Unconstr.**		**Positive**		**Pos Ord**		**Softmax**	
k	MAP	P20	MAP	P20	MAP	P20	MAP	P20	MAP	P20
2	**39.9**	**56.7**	38.7	54.7	38.8	54.8	39.9	56.6	39.9	55.8
5	39.0	53.3	38.9	54.2	39.2	54.2	40.0	55.7	**41.6**	**58.8**
10	38.0	51.4	38.5	54.5	39.9	55.3	40.1	56.1	**42.1**	**59.3**
25	36.2	48.4	35.7	52.1	37.4	52.4	40.0	55.1	**42.1**	**58.3**
50	34.3	47.0	30.4	43.9	32.9	46.4	39.0	53.4	**42.7**	**59.7**

For efficiency, for each document we only keep the distances to its k most similar documents, and set the distance to zero for all other documents. For the similarity between a query and the database we use $k = 1000$, and for the similarity between documents in the database we use $k = 200$.

6.1.2 Retrieval models with query correction terms

In our first set of experiments we compare the models presented in Section 4.1, that learn a weight vector $\mathbf{w}$ to combine the textual similarity s_t and transmedia similarity s_{vt} as defined in Eq. 1. We compare the models both with and without using the query correcting terms introduced in Section 4.2. As baseline method we follow the approach described in [1], which uses $k = 2$ neighbors to define the transmedia component and weight vector $\mathbf{w} = [1, 2]^\top$ for late fusion. We also include results obtained with this weighting but using different values for k. In Fig. 3 we show the MAP and P20 scores for each model, for different numbers of neighbors k used in the transmedia component s_{vt}.

Both in terms of MAP and P20, for all values of k, and for both classification models, the performance is significantly improved when applying the query correction terms $\{\alpha_q, \beta_q\}$. These improvements increase for larger k.

For all values of k, both with and without the query correction terms, and using both performance measures, the Comparative Classification model (CC) performs better than, or comparable to, the Relevance Classification (RC) model. The CC model also leads to slightly better performance than the baseline system of [1] that uses the manually tuned weight vector.

Best results are obtained when using relatively low values of k; depending on the performance measure and the method $k = 2$ or $k = 3$ is optimal, which is in line with [2]. In the following experiment we evaluate our learned transmedia models to assess whether there is truly nothing to be gained by using larger values of k, or whether the equal weighting is hindering performance.

6.1.3 Learned transmedia relevance models

In our second set of experiments we evaluate the transmedia weighting schemes presented in Section 3.2, defined using neighbor ranks of Eq. 2, and using the softmax function in Eq. 4. We also compare to the baseline of equal weighting, and for rank-based weights we evaluate the effect of imposing positivity and ordering constraints.

Table 3: In this table we give an overview of the best results obtained by participants to the ImageCLEF 2008 Photo Retrieval task.

Reference	MAP	P20
AVEIR [53]	31.8	43.5
UP-GPLSI [42]	33.0	43.1
DCU [43]	35.1	47.6
XRCE [1]	41.1	57.3
(our implementation)	40.1	56.4
Ours - 2 comp	42.7	59.7
Ours - 6 comp	**43.1**	**59.9**

We only present results obtained with the CC model including the correction terms, since it was outperforming the RC model in the earlier experiments. The results are summarized in Table 2.

We see that, for larger k our models substantially improve over the equal weighting baseline. For increasing values of k the performance of the baseline decreases, while that of our learned softmax weighting improves and leads to the best overall results.

It is interesting to observe that for the rank-based weighting imposing stricter constraints improves the performance, in particular for larger values of k, where a larger number of parameters needs to be estimated with a higher risk of overfitting. The softmax model, does not suffer from this problem, since it only requires a single parameter γ to be learned, and outperforms the rank-based model for $k > 2$.

In Fig. 4 we illustrate the performance of the CC model for six queries, learned with query correction terms, and softmax weighting in the transmedia relevance component s_{vt}.

6.1.4 Combining all six similarities

So far, as in [1], we have combined only single transmedia similarity, s_{vt}, with a single direct similarity s_t. Considering more terms, *e.g.* all the six similarities from Table 1, makes the manual parameter tuning much more cumbersome: there are six late fusion weights to set, and four values of k to set for the indirect similarities. Using equal weighting for all six components as in Eq. (1) with the same k value for all four pseudo-relevance terms did not lead to any improvement over the two component approach described above using any of the tested k values.

On the contrary, fixing k but learning the weight parameters with the proposed methods allowed for improvements over the learned two component models of Section 6.1.3, and this for any choice we made for k. For example, for $k = 50$ we obtain an MAP value of 43.12% and 59.87% in P20. Upon inspection we find that the learned weight for both pseudo-relevance components s_{tt} and s_{vv} equals zero, explaining partially that the improvement over the two component setting is only moderate.

6.1.5 Comparison to ImageCLEF 2008 participants

Finally, in Table 3, we compare our results to the best submissions of the ImageCLEF 2008 Photo Retrieval task. With our re-implementation of [1], using the same features

Figure 4: Example queries from the ImageCLEF Photo Retrieval Task: on the left we show the textual query and the three query images (with a blue box), on the right we show our top 10 results. Relevant images are denoted with a green box, while non-relevant images have a red box.

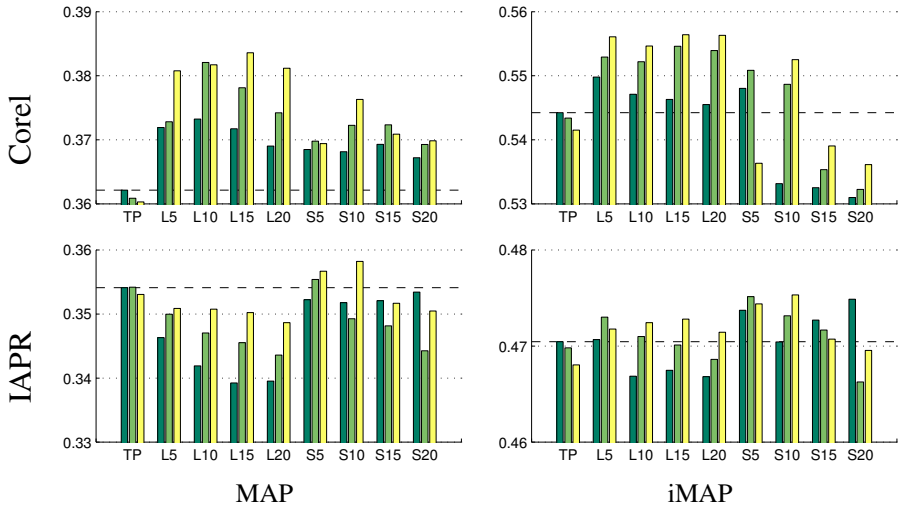


Figure 5: Image annotation performance on Corel 5K and IAPR datasets using TagProp (TP), and transmedia models using the tag distance as d^T. Usage of LTP and STP is indicated by L and S respectively, followed the value of K. Results for three different sizes of neighborhoods are shown in adjacent bars, $J = \{200, 400, 1000\}$.

and their weighting $\mathbf{w} = [1, 2]^\top$ and $k = 2$, we obtain results slightly below the ones reported by ImageCLEF in [4]. However using these features and our learning approach that integrates all six components we obtain an improvement of over 3% in both MAP and P20.

6.2 Image annotation experiments

In this section we evaluate our transmedia relevance feedback models for image annotation on two publicly available data sets: Corel 5k [16] and the IAPR TC12 [21]. Here we use TagProp [22] as a baseline for comparison. Below, we denote our linear transmedia model with LTP, *c.f*. Eq. (2), and our model based on softmax transmedia relevance as STP, *c.f*. Eq. (4).

Image auto-annotation is usually evaluated measuring the keyword based retrieval of the system. To measure this performance we use mean average precision (MAP) and break-even point precision (BEP) over keywords. MAP is obtained by computing for each keyword the average of the precisions measured after each relevant image is retrieved. BEP (or R-precision) measures for each keyword (tag) t the precision among the top n_t relevant images, where n_t is the number of images annotated with this keyword in the ground truth. However, these measures evaluate the performances of different tags independently from each other. Therefore, in order to evaluate the performance of annotating a given image with a set of labels, we propose to inverse these measures, and calculate iMAP and iBEP. These measures, instead of calculating precision over ranked images and averaging over keywords, calculate precision over ranked keywords, and average over all images.

6.2.1 Databases and feature extraction

The Corel 5K dataset contains around 5000 images, each is manually annotated for the purpose of keyword-based retrieval with between 1 and 5 keywords, out of a vocabulary of 260 words. A fixed set of 499 images is used as test, and the rest is used for training.

The IAPR TC12 dataset contains 20.000 images accompanied with textual descriptions. The annotation keywords are the common nouns of the descriptions obtained using natural language processing techniques. We use the same test and train-split, and the same vocabulary (containing the most frequent 291 words) as in [37, 22].

For a fair comparison we use the same visual features[1] as in [22]. These include the Gist descriptor [44], color histograms for RGB, LAB and HSV representations, and bag-of-word histograms computed from quantized SIFT and color descriptors. To compute the visual distances from the descriptors we follow [37, 22] and use ℓ_2 as the base metric for Gist, ℓ_1 for color histograms, and χ^2 for the others. Besides these collection of 15 descriptors we often use them equally weighted and averaged and refer to that as JEC (short for Joint Equal Contribution) distance. Unless specified otherwise, use JEC as the single visual distance in most of our experiments.

As textual distance we use two different distances, one for set of tags and another for language model based text representations where the latter features are available. We define the tag distance based on the training labels as the intersection-over-union measure over the ground truth annotations of training images:

$$d_t(k,j) = 1 - |Y_k \cap Y_j| / |Y_k \cup Y_j|, \tag{26}$$

with $Y_k = \{t | y_{kt} = +1\}$. For the IAPR data set we also consider using the image captions to compute the cross-entropy measure based on a language model, see Section 6.1.1. However, since the tags in IAPR are the extracted nouns from the captions, as expected the performances with tag and text distances are similar (see results in Section 6.2.3 and Section 6.2.4).

6.2.2 Annotation with transmedia relevance models

In Fig. 5 we show the MAP and iMAP performance of our proposed transmedia pseudo-relevance models (LTP and STP) compared to the original TagProp, on the Corel and IAPR datasets. For the transmedia distance d_{vt} we use the intersection-over-union tag distance as the (second) textual distance.

The results on the Corel dataset (Fig. 5 upper row) show that for most parameter configurations and for both performance measures we significantly outperform the TagProp baseline. When comparing TagProp (using J=200, MAP 36.2%) with our method (using LTP, K=15, J=1000, MAP 38.4%), we see that for 144(/26/90) of the 260 keywords our method has as average precision that is higher (/equal/lower) than TagProp. Furthermore, the figures show that LTP generally outperforms STP on this data set. Finally, if we increase the neighborhood size J (indicated by different bars) the performances increases in our case while for TagProp slightly decreases.

The results on the IAPR dataset (Fig. 5 lower row) show that using the transmedia relevance feedback models for this database do not yield much higher performance than TagProp. Another difference compared to the Corel dataset is that on this dataset the STP model clearly outperforms the LTP model.

[1] Available for download from `http://lear.inrialpes.fr/data`.

Table 4: Performance of different distances d^t in the second step of pseudo and transmedia relevance models, using $J = 1000$, and $K = 20$.

		LTP				STP			
		MAP	BEP	iMAP	iBEP	MAP	BEP	iMAP	iBEP
Corel 5K	TagProp	36.0	32.5	54.2	47.6				
	$d_t = \{\text{Jec}\}$	36.0	32.5	54.2	47.8	36.0	32.5	54.2	**47.8**
	$d_t = \{\text{Tag}\}$	**38.1**	33.8	**55.6**	49.3	**37.0**	**33.1**	53.6	47.0
	$d_t = \{\text{Jec, Tag}\}$	37.9	**33.9**	55.5	**49.7**	36.6	32.9	**53.7**	47.2
		LTP				STP			
		MAP	BEP	iMAP	iBEP	MAP	BEP	iMAP	iBEP
IAPR TC 12	TagProp	**35.4**	**36.0**	47.0	**42.6**				
	$d_t = \{\text{Jec}\}$	35.1	**36.0**	46.7	42.2	35.1	36.0	46.7	42.3
	$d_t = \{\text{Tag}\}$	34.7	35.5	47.1	42.3	35.6	36.3	47.4	42.7
	$d_t = \{\text{Text}\}$	34.9	35.9	**47.5**	42.2	**35.9**	36.3	**48.0**	42.8
	$d_t = \{\text{Jec,Tag}\}$	34.5	35.5	46.8	41.9	35.3	36.3	47.1	42.0
	$d_t = \{\text{Jec,Text}\}$	34.5	35.7	47.1	42.0	35.5	36.3	47.5	42.8
	$d_t = \{\text{Tag, Text}\}$	34.7	35.8	47.2	42.1	35.7	**36.5**	47.9	**43.0**

Table 5: Combining four visual distances with TagProp and in its transmedia extensions, using $J = 400$, and $K = 10$

		LTP				STP			
		MAP	BEP	iMAP	iBEP	MAP	BEP	iMAP	iBEP
IAPR TC 12	TagProp	35.7	36.1	49.0	44.1				
	$d_t = \{\text{Jec}\}$	35.0	35.3	48.6	44.1	35.0	35.6	48.6	44.0
	$d_t = \{\text{Tag}\}$	36.0	**36.7**	49.6	44.6	35.6	36.1	49.2	44.4
	$d_t = \{\text{Text}\}$	**36.4**	**36.7**	49.6	44.3	35.7	35.7	49.5	44.2
	$d_t = \{\text{Tag, Text}\}$	36.2	36.6	**49.9**	**44.8**	**35.8**	**36.6**	**49.8**	**44.6**

6.2.3 Comparing pseudo and transmedia relevance

In Table 4 we further compare the performance of the LTP and STP with different choices for the distance d_t that is used in the second step of the pseudo and transmedia relevance models. We use (a) the visual distance (JEC), generating a visual pseudo-relevance feedback model, (b) the tag distance, (c) the cross-entropy based text distance (for the IAPR dataset), and (d) a combination of the previous.

On the Corel dataset, using visual pseudo-relevance feedback performs similarly as the baseline TagProp, the transmedia model (using the tag distance) clearly improves the retrieval and annotation performance, and the combination of the two performs comparable as using just the transmedia model. Just as in Fig. 5, LTP seems to outperform STP for this database and these settings.

For the IAPR dataset, we obtain the highest scores $d_t =$ Text, improving around $.5\%$ the retrieval scores, and up to 1% the iMAP. Comparing on the AP per keyword (291 in total), our method (using text distance, STP, J=400,K=10) outperforms/equals/underperforms TagProp (using J=200) in 168/3/120 cases.

6.2.4 Learning visual distance weights

In our final experiment, instead of using the visual JEC distance, we selected the four visual distances that obtain the highest weights when learning TagProp with the 15 individual feature distances (see [54] for an overview of the learned weights). We define a transmedia relevance models using each of them and learn their combination weights with the proposed method. Note that in this case LTP and STP use both the four visual distances as well as the four transmedia distances defined with them.

In Table 5 we compare the results of TagProp (only using the four visual distances), with the LTP and STP models that also include the transmedia distances defined with the four visual distances. In this case the LTP model performs slightly better than the STP model. Using LTP with either the Tag or Text distance to define the transmedia distance we obtain modest improvements on all performance measures.

7 Conclusion

In this paper we propose two parameterizations of use transmedia pseudo-relevance feedback models, that generalize the models proposed in [1]. Our models extend the latter by incorporating a weighting among the neighbors used to compute the transmedia pseudo-relevance feedback score. The parameters of the transmedia pseudo-relevance feedback components are learned from data. Similarly, we learn late fusion weights that combine the transmedia pseudo-relevance feedback components with other similarity measures. We applied these models for multimodal retrieval and image annotation.

A second contribution of this paper is the introduction of multiplicative and additive correction terms to learn multimodal retrieval score functions. The motivation for this is that while ranking performance is invariant to such terms, the objective functions of the learning algorithms are not. During model training the correction terms are optimized along with the actual model parameters, ensuring that the learning algorithm will not return a suboptimal score function due to inter-query differences in the distribution of similarity values.

We evaluated our retrieval models using the data from the ImageCLEF 2008 Photo Retrieval task. The results showed that our models, using our transmedia pseudo-relevance feedback components and learned using the correction terms, outperform the best results know to us for this problem which are based on manually tuned transmedia relevance models.

For image annotation, we integrated our transmedia pseudo-relevance feedback components as additional distances in the TagProp model [22]. Our experimental results show that on the Corel 5K dataset, we significantly improve the state-of-the-art results of TagProp. On the more challenging IAPR TC12 dataset, we obtain more modest improvements of up to 1% over TagProp. These results show that using the textual information associated with training images can improve auto-annotation of images for which no textual information is available.

In future work we would like to learn our transmedia pseudo-relevance feedback models using alternative learning methods, such as [58], that are more directly related to the performance measures we are interested in.

References

[1] J. Ah-pine, C. Cifarelli, S. Clinchant, G. Csurka, and J.-M. Renders. XRCE's participation to ImageCLEF 2008. In *Working Notes of the CLEF Workshop*. CLEF Campaign, 2008.

[2] J. Ah-Pine, S. Clinchant, G. Csurka, F. Perronnin, and J-M. Renders. Leveraging image, text and cross-media similarities for diversity-focused multimedia retrieval. In H. Müller, P. Clough, T. Deselaers, and B. Caputo, editors, *ImageCLEF - Experimental Evaluation in Visual Information Retrieval*. Springer, 2010.

[3] N. Ailon. A simple linear ranking algorithm using query dependent intercept variables, 2009.

[4] T. Arni, P. Clough, M. Sanderson, and M. Grubinger. Overview of the ImageCLEFphoto 2008 photographic retrieval task. In *Working Notes of the CLEF Workshop*. CLEF Campaign, 2008.

[5] K. Barnard, P. Duygulu, D. Forsyth, N. de Freitas, D. Blei, and M. Jordan. Matching words and pictures. *JMLR*, 3:1107–1135, 2003.

[6] R. Bekkerman and J. Jeon. Multi-modal clustering for multimedia collections. In *CVPR*. IEEE, 2007.

[7] T. Berg, A. Berg, J. Edwards, M. Maire, R. White, Y. Teh, E. Learned-Miller, and D. Forsyth. Names and faces in the news. In *CVPR*, 2004.

[8] Chris Burges, Tal Shaked, Erin Renshaw, Ari Lazier, Matt Deeds, Nicole Hamilton, and Greg Hullender. Learning to rank using gradient descent. In *ICML*, pages 89–96, New York, NY, USA, 2005. ACM.

[9] G. Carneiro, A. Chan, P. Moreno, and N. Vasconcelos. Supervised learning of semantic classes for image annotation and retrieval. *PAMI*, 2007.

[10] Y.-C. Chang and H.-H. Chen. Approaches of using a word-image ontology and an annotated image corpus as intermedia for cross-language image retrieval. In *Working Notes of the CLEF Workshop*, 2006.

[11] G. Chechik, V. Sharma, U. Shalit, and S. Bengio. Large scale online learning of image similarity through ranking. *JMLR*, 11:1109–1135, 2010.

[12] O. Chum, J. Philbin, J. Sivic, M. Isard, and A. Zisserman. Total recall: Automatic query expansion with a generative feature model for object retrieval. In *ICCV*, 2007.

[13] S. Clinchant, J.-M. Renders, and G. Csurka. XRCE's participation to ImageCLEFphoto 2007. In *Working Notes of the CLEF Workshop*, 2007.

[14] C. Cusano, G. Ciocca, and R. Schettini. Image annotation using SVM. In *Proceedings Internet imaging (SPIE)*, 2004.

[15] A. Depeursinge and H. Müller. Fusion techniques for combining textual and visual information retrieval. In H. Müller, P. Clough, T. Deselaers, and B. Caputo, editors, *ImageCLEF - Experimental Evaluation in Visual Information Retrieval*. Springer, 2010.

[16] P. Duygulu, K. Barnard, N. de Freitas, and D. Forsyth. Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary. In *ECCV*, 2002.

[17] M. Everingham, L. Van Gool, C. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2007 Results. http://www.pascal-network.org/challenges/VOC/voc2007/workshop.

[18] S. Feng, R. Manmatha, and V. Lavrenko. Multiple Bernoulli relevance models for image and video annotation. In *CVPR*, 2004.

[19] A Frome, Jitendra Malik, and Y. Singer. Image retrieval and classification using local distance functions. In *NIPS*, volume 19, pages 417–424, 2007.

[20] D. Grangier and S. Bengio. A discriminative kernel-based model to rank images from text queries. *PAMI*, 2008.

[21] M. Grubinger, P. Clough, H. Müller, and T. Deselaers. The IAPR benchmark: A new evaluation resource for visual information systems. In *Int. Conf. on Language Resources and Evaluation*, 2006.

[22] M. Guillaumin, T. Mensink, J. Verbeek, and C. Schmid. Tagprop: Discriminative metric learning in nearest neighbor models for image auto-annotation. In *ICCV*, 2009.

[23] M. Guillaumin, T. Mensink, J. Verbeek, and C. Schmid. Face recognition from caption-based supervision. *IJCV*, 2011. to appear.

[24] T. Hertz, A. Bar-Hillel, and D. Weinshall. Learning distance functions for image retrieval. In *CVPR*, 2004.

[25] H. Jégou, M. Douze, and C. Schmid. Packing bag-of-features. In *ICCV*, 2009.

[26] J. Jeon, V. Lavrenko, and R. Manmatha. Automatic image annotation and retrieval using cross-media relevance models. In *ACM SIGIR*, 2003.

[27] T. Joachims. A support vector method for multivariate performance measures. In *ICML*, 2005.

[28] J. Kludas, S. Marchand-Maillet, and E. Bruno. Information fusion in multimedia information retrieval. In *Workshop on Adaptive Multimedia Retrieval*, 2007.

[29] J. Krapac, M. Allan, J. Verbeek, and F. Jurie. Improving web-image search results using query-relative classifiers. In *CVPR*, 2010.

[30] I. Laptev, M. Marszałek, C. Schmid, and B. Rozenfeld. Learning realistic human actions from movies. In *CVPR*, 2008.

[31] V. Lavrenko, R. Manmatha, and J. Jeon. A model for learning the semantics of pictures. In *NIPS*, 2003.

[32] D. Lewis. Applying support vector machines to the TREC-2001 batch filtering and routing tasks. In *Proceedings of (TREC)*, 2001.

[33] H. Li. *Learning to Rank for Information Retrieval and Natural Language Processing*. Morgan & Claypool Publishers, 2011.

[34] X. Li, L. Chen, L. Zhang, F. Lin, and W. Ma. Image annotation by large-scale content based image retrieval. In *ACM Multimedia*, 2006.

[35] J. Liu, M. Li, Q. Liu, H. Lu, and S. Ma. Image annotation via graph learning. *Pattern Recognition*, 2009.

[36] N. Maillot, J.-P. Chevallet, V. Valea, and J. Lim. IPAL inter-media pseudo-relevance feedback approach to ImageCLEF 2006. In *Working Notes of the CLEF Workshop*, 2006.

[37] A. Makadia, V. Pavlovic, and S. Kumar. A new baseline for image annotation. In *ECCV*, 2008.

[38] T. Mensink, G. Csurka, F. Perronnin, J. Sánchez, and J. Verbeek. LEAR and XRCE's participation to visual concept detection task. In *Working Notes of the CLEF Workshop*, 2010.

[39] T. Mensink, G. Csurka, and J. Verbeek. Trans media relevance feedback for image autoannotation. In *BMVC*, 2010.

[40] F. Monay and D. Gatica-Perez. PLSA-based image auto-annotation: Constraining the latent space. In *ACM Multimedia*, 2004.

[41] H. Müller, P. Clough, T. Deselaers, and B. Caputo. *ImageCLEF - Experimental Evaluation in Visual Information Retrieval*. Springer, 2010.

[42] S. Navarro, M. García, F. Llopis, M Díaz, R. Muñoz, M. Martín, L. Ureña, and A. Montejo. Text-mess in the ImageCLEFphoto08 task. In *Working Notes of the CLEF Workshop*, 2008.

[43] N. O'Hare, P. Wilkins, C. Gurrin, E. Newman, G. Jones, and A. Smeaton. Dcu at imageclefphoto 2008. In *Working Notes of the CLEF Workshop*, 2008.

[44] A. Oliva and A. Torralba. Modeling the shape of the scene: a holistic representation of the spatial envelope. *IJCV*, 2001.

[45] J. Pan, H. Yang, C. Faloutsos, and P. Duygulu. Automatic multimedia cross-modal correlation discovery. In *ACM SIGKDD*, 2004.

[46] F Perronnin and C Dance. Fisher kernels on visual vocabularies for image categorization. In *CVPR*, 2007.

[47] F. Perronnin, Yan Liu, and J.-M. Renders. A family of contextual measures of similarity between distributions with application to image retrieval. In *CVPR*, 2009.

[48] J. Ponte and W. Croft. A language modelling approach to information retrieval. In *SIGIR*, 1998.

[49] Tao Qin, Tie-Yan Liu, Jun Xu, and Hang Li. Letor: A benchmark collection for research on learning to rank for information retrieval. *Information Retrieval*, 13:346–374, 2010.

[50] Gerard Salton and Chris Buckley. Improving retrieval performance by relevance feedback. *J. of the American Society for Information Science*, 1990.

[51] F. Schroff, A. Criminisi, and A. Zisserman. Harvesting image databases from the web. In *ICCV*, 2007.

[52] C. Snoek, M. Worring, and A. Smeulders. Early versus late fusion in semantic video analysis. *ACM Multimedia*, 2005.

[53] S. Tollari, M. Detyniecki, M. Ferecatu, H. Glotin, P. Mulhem, M. Amini, A. Fakeri-Tabrizi, P. Gallinari, H. Sahbi, and Z. Zhao. Consortium AVEIR at imageCLEF photo 2008: on the fusion of runs. In *Working Notes of the CLEF Workshop*, 2008.

[54] J. Verbeek, M. Guillaumin, T. Mensink, and C. Schmid. Image Annotation with TagProp on the MIRFLICKR set. *ACM Multimedia Information Retrieval*, 2010.

[55] X. Wang, L. Zhang, F. Jing, and W.-Y. Ma. AnnoSearch: Image auto-annotation by search. In *CVPR*, 2006.

[56] J. Weston, S. Bengio, and N. Usunier. Large scale image annotation: Learning to rank with joint word-image embeddings. In *ECML*, 2010.

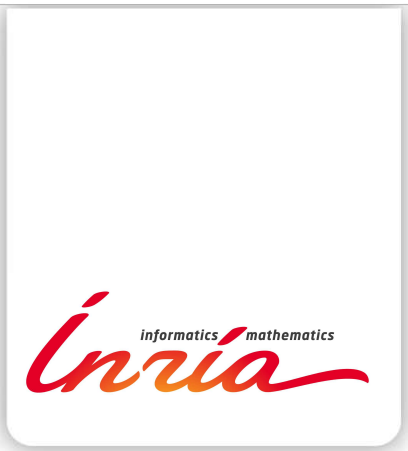
[57] O. Yakhnenko and V. Honavar. Annotating images and image objects using a hierarchical Dirichlet process model. In *Workshop on Multimedia Data Mining - SIGKDD*, 2008.

[58] Y. Yue, T. Finley, F. Radlinski, and T. Joachims. A support vector method for optimizing average precision. In *SIGIR*, 2007.

[59] H. Zhang, A. Berg, M. Maire, and J. Malik. SVM-KNN: Discriminative nearest neighbor classification for visual category recognition. In *CVPR*, 2006.

[60] Z. Zheng, H. Zha, T. Zhang, O. Chapelle, K. Chen, and G. Sun. A general boosting method and its application to learning ranking functions for web search. In *NIPS*, 2008.

Contents



RESEARCH CENTRE
GRENOBLE – RHÔNE-ALPES

Inovallée
655 avenue de l'Europe Montbonnot
38334 Saint Ismier Cedex

Publisher
Inria
Domaine de Voluceau - Rocquencourt
BP 105 - 78153 Le Chesnay Cedex
inria.fr

ISSN 0249-0803